

## EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1	(garbage adj1 collection) same snapshot same (indicat\$3 near6 state)	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:26
L2	2	restor\$3 near4 state with thread same (garbage near2 collection)	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:28
L3	16	restor\$3 near4 state with thread and (garbage near2 collection)	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:34
L4	28092	711/???.ccls.	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:34
L5	126	l4 and (synchronization) and (garbage near2 collection)	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:34
L6	4	l5 and (snapshot) and restor\$3	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:36
L7	12	consistent near2 state same (garbage adj1 collection)	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:42
L8	3	consistent adj1 point with synchronization	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:43
L9	10	consistent adj1 point same synchronization	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:44
L10	672	711/152.ccls.	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:44
L11	14	l10 and (garbage near2 collection)	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:51

## EAST Search History

L12	10	(snapshot cache) with thread with state same restor\$3	US-PGPUB; USPAT; USOCR; EPO	OR	ON	2006/04/03 09:54
-----	----	---	--------------------------------------	----	----	------------------

[Home](#) | [Login](#) | [Logout](#) | [Access Information](#) | [Alerts](#) |

Welcome United States Patent and Trademark Office

[Search Session History](#)[BROWSE](#)[SEARCH](#)[IEEE XPLORE GUIDE](#)

Edit an existing query or  
compose a new query in the  
Search Query Display.

Mon, 3 Apr 2006, 10:01:06 AM EST

## Search Query Display

Select a search number (#)  
to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

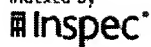


## Recent Search Queries

- #1 ( ( request consistent state<in>metadata ) <and> ( garbage collection<in>metadata ) )
- #2 ( ( garbage collection<in>metadata ) <and> ( restore state<in>metadata ) )
- #3 ( ( garbage collection<in>metadata ) <and> ( synchronization<in>metadata ) )
- #4 ( ( garbage collection<in>metadata ) <and> ( synchronization<in>metadata ) )



Indexed by

[Help](#) [Contact Us](#) [Privacy &](#)

© Copyright 2006 IEEE -


[Home](#) | [Login](#) | [Logout](#) | [Access Information](#) | [Alerts](#) |

Welcome United States Patent and Trademark Office

☐ Search Results

BROWSE

SEARCH

IEEE XPLORE GUIDE

Results for "( ( garbage collection&lt;in&gt;metadata ) &lt;and&gt; ( synchronization&lt;in&gt;metadata ) )"

e-mail

Your search matched 19 of 1332769 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

## » Search Options

[View Session History](#)
[New Search](#)

## Modify Search

☐ Check to search only within this results set
Display Format: ☐ Citation ☒ Citation & Abstract

## » Key

IEEE JNL IEEE Journal or Magazine

IEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

[Select All](#) [Deselect All](#)

- ☐ 1. **Lock-free garbage collection for multiprocessors**  
 Herlihy, M.P.; Moss, J.E.B.;  
[Parallel and Distributed Systems, IEEE Transactions on](#)  
 Volume 3, Issue 3, May 1992 Page(s):304 - 311  
 Digital Object Identifier 10.1109/71.139204  
**Summary:** Garbage collection algorithms for shared-memory multiprocessors  
 some form of global synchronization to preserve consistency. Such global sync  
 lead to problems on asynchronous architectures: if one process is halted or....  
[AbstractPlus](#) | [Full Text: PDF\(732 KB\)](#) IEEE JNL  
[Rights and Permissions](#)
- ☐ 2. **Lazy garbage collection of recovery state for fault-tolerant distributed sh:**  
 Sultan, F.; Nguyen, T.D.; Iftode, L.;  
[Parallel and Distributed Systems, IEEE Transactions on](#)  
 Volume 13, Issue 7, July 2002 Page(s):673 - 686  
 Digital Object Identifier 10.1109/TPDS.2002.1019857  
**Summary:** In this paper, we address the problem of garbage collection in a sir  
 tolerant home-based lazy release consistency (HLRC) distributed shared-mem  
 system based on independent checkpointing and logging. Our solution uses la:  
[AbstractPlus](#) | [References](#) | [Full Text: PDF\(1287 KB\)](#) IEEE JNL  
[Rights and Permissions](#)
- ☐ 3. **An on-chip garbage collection coprocessor for embedded real-time syste**  
 Meyer, M.;  
[Embedded and Real-Time Computing Systems and Applications, 2005. Proce](#)  
[International Conference on](#)  
 17-19 Aug. 2005 Page(s):517 - 524  
 Digital Object Identifier 10.1109/RTCSA.2005.25  
**Summary:** Garbage collection considerably increases programmer productivit  
 quality. However, it is difficult to implement garbage collection both efficien  
 a real-time systems. Today, garbage collection is exclusively realized in.....  
[AbstractPlus](#) | [Full Text: PDF\(136 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
- ☐ 4. **A fast analysis for thread-local garbage collection with dynamic class loa**  
 Jones, R.; King, A.C.;  
[Source Code Analysis and Manipulation, 2005. Fifth IEEE International Works](#)  
 30 Sept.-1 Oct. 2005 Page(s):129 - 138  
 Digital Object Identifier 10.1109/SCAM.2005.1

**Summary:** Long-running, heavily multi-threaded, Java server applications make demands of garbage collector (GC) performance. Synchronisation of all applications before garbage collection is a significant bottleneck for JVMs that use native threads.

[AbstractPlus](#) | [Full Text: PDF\(296 KB\)](#) IEEE CNF

[Rights and Permissions](#)

5. **Distributed garbage collection of active objects**  
 Washabaugh, D.M.; Kafura, D.;  
[Distributed Computing Systems, 1991., 11th International Conference on 20-24 May 1991](#) Page(s):369 - 376  
 Digital Object Identifier 10.1109/ICDCS.1991.148691  
**Summary:** Distributed automatic garbage collection of objects possessing their own control is discussed. The relevance of garbage collection and concurrent object applications is briefly discussed, and the specific model of concurrent garbage collection is described.  
[AbstractPlus](#) | [Full Text: PDF\(548 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
6. **Garbage collection in a distributed object-oriented system**  
 Gupta, A.; Fuchs, W.K.;  
[Knowledge and Data Engineering, IEEE Transactions on Volume 5, Issue 2, April 1993](#) Page(s):257 - 265  
 Digital Object Identifier 10.1109/69.219734  
**Summary:** An algorithm for garbage collection in distributed systems with object processor boundaries is described. The algorithm allows local garbage collection in the system to proceed independently of local collection at the other processors.  
[AbstractPlus](#) | [Full Text: PDF\(820 KB\)](#) IEEE JNL  
[Rights and Permissions](#)
7. **Lazy garbage collection of recovery state for fault-tolerant distributed shared-memory systems**  
 Sultan, F.; Nguyen, T.D.; Ifode, L.;  
[Parallel and Distributed Systems, IEEE Transactions on Volume 13, Issue 10, Oct. 2002](#) Page(s):1085 - 1098  
 Digital Object Identifier 10.1109/TPDS.2002.1041885  
**Summary:** We address the problem of garbage collection in a single-failure fault-tolerant based lazy release consistency (HLRC) distributed shared-memory (DSM) system. Our solution uses independent checkpointing and logging. Our solution uses laziness in garbage collection.  
[AbstractPlus](#) | [References](#) | [Full Text: PDF\(1399 KB\)](#) IEEE JNL  
[Rights and Permissions](#)
8. **Distributed garbage collection by timeouts and backward inquiry**  
 Sung-Wook Ryu; Eul Gyu Im; Neuman, B.C.;  
[Computer Software and Applications Conference, 2003. COMPSAC 2003. Proceedings. 3-6 Nov. 2003](#) Page(s):426 - 432  
**Summary:** We present a practical and efficient garbage collection mechanism for distributed systems. The mechanism collects all garbage including distributed garbage without global synchronization or backward links. The primary method used for garbage collection is based on timeouts and backward inquiry.  
[AbstractPlus](#) | [Full Text: PDF\(268 KB\)](#) IEEE CNF  
[Rights and Permissions](#)
9. **An analysis of the garbage collection performance in Sun's HotSpot™ JVM**  
 Dykstra, L.; Srisa-an, W.; Chang, J.M.;  
[Performance, Computing, and Communications Conference, 2002. 21st IEEE 3-5 April 2002](#) Page(s):335 - 339  
 Digital Object Identifier 10.1109/IPCCC.2002.995167  
**Summary:** Sun Microsystems introduced the Java HotSpot Virtual Machine as a performance engine for the Java programming language. The improvements of the Java Virtual Machine include dynamic compilation and optimization, better thread scheduling, and better garbage collection.  
[AbstractPlus](#) | [Full Text: PDF\(563 KB\)](#) IEEE CNF

[Rights and Permissions](#)

10. **Real-time garbage collection in multi-threaded systems on a single processor**  
 Siebert, F.;  
[Real-Time Systems Symposium, 1999. Proceedings. The 20th IEEE](#)  
 1-3 Dec. 1999 Page(s):277 - 278  
 Digital Object Identifier 10.1109/REAL.1999.818854  
**Summary:** We show the difficulties that arise for the implementation of a real-time garbage collector (GC) in a multi-threaded system. A mechanism for synchronization between threads is proposed for a single processor system. It is shown how this mechanism can be implemented.  
[AbstractPlus](#) | Full Text: [PDF](#)(20 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
11. **A framework of secure object system architecture**  
 Rousseau, L.; Natkin, S.;  
[Object-Oriented Real-Time Dependable Systems, 1997. Proceedings., Third International Workshop on](#)  
 5-7 Feb. 1997 Page(s):108 - 115  
 Digital Object Identifier 10.1109/WORDS.1997.609932  
**Summary:** The paper presents some basic principles for the design of secure operating systems. The security relies on the control of right to call an object method. Capabilities are used to implement the control scheme. Our capabilities are named "secure".  
[AbstractPlus](#) | Full Text: [PDF](#)(632 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
12. **Conservative garbage collection on distributed shared memory systems**  
 Weimin Yu; Cox, A.;  
[Distributed Computing Systems, 1996., Proceedings of the 16th International Conference on](#)  
 27-30 May 1996 Page(s):402 - 410  
 Digital Object Identifier 10.1109/ICDCS.1996.507988  
**Summary:** In this paper we present the design and implementation of a conservative garbage collection algorithm for distributed shared memory (DSM) applications that use languages like C or C++, and evaluate its performance. In the absence of language support, the algorithm is implemented in C.  
[AbstractPlus](#) | Full Text: [PDF](#)(820 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
13. **Distributed/concurrent garbage collection in distributed shared memory**  
 Kordale, R.; Ahamad, M.; Shilling, J.;  
[Object Orientation in Operating Systems, 1993., Proceedings of the Third International Workshop on](#)  
 9-10 Dec. 1993 Page(s):51 - 60  
 Digital Object Identifier 10.1109/IWOOS.1993.324927  
**Summary:** We present a distributed and concurrent garbage collection algorithm suited for distributed shared memory systems. Traversal of objects during garbage collection happens "in place" in the sense that the objects are not moved.  
[AbstractPlus](#) | Full Text: [PDF](#)(792 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
14. **A parallel asynchronous garbage collection algorithm for distributed systems**  
 Bagherzadeh, N.; Heng, S.; Wu, C.;  
[Knowledge and Data Engineering, IEEE Transactions on](#)  
 Volume 3, Issue 1, March 1991 Page(s):100 - 107  
 Digital Object Identifier 10.1109/69.75893  
**Summary:** The problem of distributed garbage collection is discussed. An algorithm for distributed asynchronous garbage collection is presented. The liveness and safety of this method are demonstrated. The algorithm does not require a global state.  
[AbstractPlus](#) | Full Text: [PDF](#)(668 KB) [IEEE JNL](#)  
[Rights and Permissions](#)

15. **Stampede: a cluster programming middleware for interactive stream-oriented data processing**

- applications  
Ramachandran, U.; Nikhil, R.S.; Rehg, J.M.; Angelov, Y.; Paul, A.; Adhikari, S.  
K.M.; Harel, N.; Knobe, K.;  
[Parallel and Distributed Systems, IEEE Transactions on](#)  
Volume 14, Issue 11, Nov. 2003 Page(s):1140 - 1154  
Digital Object Identifier 10.1109/TPDS.2003.1247674  
**Summary:** Emerging application domains such as interactive vision, animation  
collaboration display dynamic scalable parallelism and high-computational req  
making them good candidates for executing on parallel architectures such as {  
[AbstractPlus](#) | [References](#) | Full Text: [PDF](#)(1235 KB) [IEEE JNL](#)  
[Rights and Permissions](#)
16. **Optimal Asynchronous Garbage Collection for RDT Checkpointing Proto**  
Schmidt, R.; Garcia, I.C.; Pedone, F.; Buzato, L.E.;  
[Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE I](#)  
[Conference on](#)  
06-10 June 2005 Page(s):167 - 176  
Digital Object Identifier 10.1109/ICDCS.2005.58  
**Summary:** Communication-induced checkpointing protocols that ensure rollba  
trackability (RDT) guarantee important properties to the recovery system witho  
coordination. However, to the best of our knowledge, there was no garbage co  
[AbstractPlus](#) | Full Text: [PDF](#)(216 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
17. **Asynchronous Complete Distributed Garbage Collection**  
Veiga, L.; Ferreira, P.;  
[Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE](#)  
04-08 April 2005 Page(s):24a - 24a  
Digital Object Identifier 10.1109/IPDPS.2005.113  
**Summary:** Most Distributed Garbage Collection (DGC) algorithms are not con  
to reclaim distributed cycles of garbage. Those that achieve such a level of cor  
very costly as they require either some kind of synchronization or consen.....  
[AbstractPlus](#) | Full Text: [PDF](#)(264 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
18. **Dead timestamp identification in Stampede**  
Harel, N.; Mandviwala, H.A.; Knobe, K.; Ramachandran, U.;  
[Parallel Processing, 2002. Proceedings. International Conference on](#)  
18-21 Aug. 2002 Page(s):101 - 108  
Digital Object Identifier 10.1109/ICPP.2002.1040864  
**Summary:** Stampede is a parallel programming system to support computatio  
applications including interactive vision, speech and multimedia collaboration.  
alleviates concerns such as communication, synchronization, and buffer mana  
[AbstractPlus](#) | Full Text: [PDF](#)(478 KB) [IEEE CNF](#)  
[Rights and Permissions](#)
19. **A fast parallel conservative garbage collector for concurrent object-orient**  
Matsuoka, S.; Furuso, S.; Yonezawa, A.;  
[Object Orientation in Operating Systems, 1991. Proceedings., 1991 Internation](#)  
17-18 Oct. 1991 Page(s):87 - 93  
Digital Object Identifier 10.1109/IWOOS.1991.183027  
**Summary:** In future OO-OS, multiple applications written in heterogeneous lan  
interact via shared objects. There, conservative GC (garbage collection) could  
management independent of programming languages and applications. Conse  
[AbstractPlus](#) | Full Text: [PDF](#)(540 KB) [IEEE CNF](#)  
[Rights and Permissions](#)